

## WEST Search History

[Hide Items](#) | [Restore](#) | [Clear](#) | [Cancel](#)

DATE: Thursday, August 17, 2006

<a href="#">Hide?</a>	<a href="#">Set Name</a>	<a href="#">Query</a>	<a href="#">Hit Count</a>
<i>DB=PGPB,USPT,USOC; PLUR=NO; OP=OR</i>			
<input type="checkbox"/>	L66	L64 and (encod\$ near (bit or bits))	17
<input type="checkbox"/>	L65	L64 and (encod\$ near lock\$)	1
<input type="checkbox"/>	L64	(l58 or l59 or l60 or l61 or l62 or l63) and ((virtual address) with (bit or bits) with (object or objects or class or table or tables))	254
<input type="checkbox"/>	L63	707/103r-103z.ccls.	1701
<input type="checkbox"/>	L62	707/100.ccls.	4568
<input type="checkbox"/>	L61	707/8.ccls.	1060
<input type="checkbox"/>	L60	711/100.ccls.	1406
<input type="checkbox"/>	L59	711/6.ccls.	198
<input type="checkbox"/>	L58	711/2.ccls.	336
<input type="checkbox"/>	L57	L56 and address.ti.	44
<input type="checkbox"/>	L56	((bit or bits) near lock)	1726
<input type="checkbox"/>	L55	(two adj1 bit adj1 lock)	5
<input type="checkbox"/>	L54	(two-bit adj1 lock)	3
<input type="checkbox"/>	L53	(two-bit lock)	738242
<input type="checkbox"/>	L52	L49 and (lock near (bit or bits))	4
<input type="checkbox"/>	L51	L49 and (lock with (bit or bits))	11
<input type="checkbox"/>	L50	L49 and lock	63
<input type="checkbox"/>	L49	L20 and address.ti.	1762
<input type="checkbox"/>	L48	L47 and address.ti.	0
<input type="checkbox"/>	L47	L20 and (lock with encode)	40
<input type="checkbox"/>	L46	L45 and (word or words)	1
<input type="checkbox"/>	L45	6987813.pn.	1
<input type="checkbox"/>	L44	L43 and address.ti.	35
<input type="checkbox"/>	L43	L20 and (address same word same n same (bit or bits))	1252
<input type="checkbox"/>	L42	L41 and virtual	11
<input type="checkbox"/>	L41	L40 and encod\$	51
<input type="checkbox"/>	L40	L38 and L39	218
<input type="checkbox"/>	L39	address.ti.	6442
<input type="checkbox"/>	L38	(address same word same n same (bit or bits))	6799

10/19/01, 230

<input type="checkbox"/>	L37 L36 and (n near (bit or bits))	0
<input type="checkbox"/>	L36 (virtual with encode with (bit or bits) with address)	4
<input type="checkbox"/>	L35 ((object or objects) with virtual with encode with (bit or bits) with address)	0
<input type="checkbox"/>	L34 L32 and memory	20
<input type="checkbox"/>	L33 L32 and dmemory	14764
<input type="checkbox"/>	L32 L31 and (pointer or pointers)	23
<input type="checkbox"/>	L31 L30 and encod\$	67
<input type="checkbox"/>	L30 L28 and header.ab.	124
<input type="checkbox"/>	L29 L28 and header.ti.	11
<input type="checkbox"/>	L28 L20 and (address near (bit or bits))	9366
<input type="checkbox"/>	L27 L25 and (address near k-bit)	1
<input type="checkbox"/>	L26 L24 and L25	1
<input type="checkbox"/>	L25 L20 and (encod\$ near lock\$)	78
<input type="checkbox"/>	L24 L23 and (pointer or pointers)	2
<input type="checkbox"/>	L23 L22 and lock	2
<input type="checkbox"/>	L22 L21 and memory	15
<input type="checkbox"/>	L21 L20 and (address near k-bit)	15
<i>DB=PGPB; PLUR=NO; OP=OR</i>		

2 5551952 5597190 5633515 5666158 5804748 5850927 6012554 6043646  
7071869 20060087473 6418140 6633592 5539871 6078931 6085196 4294543  
4777978 5347629 5360214 5685776 5835591 5893798 6334212 6424354  
20060082597 6098185 6278838 6341198 5600768 6324178 6334213 6442752  
20020131408 20020167948 6907437 6138202 6135653 5002479 5467472  
5680616 4432170 4498023 4992931 5339312 5430839 5432937 5495509  
5604905 5610660 5617455 5729755 5754564 5768447 5819283 5818447  
5852664 5974548 6175862 6199100 6249319 6249788 6317500 6351815  
5649218 5834040 6026192 6359910 20040124158 20050115911 6147944  
6400889 6173258 6475871 20060107175 5710499 6134627 5900001 5920876  
5687368 5761510 5832268 5970242 6223344 6223344 6237043 4772882  
4939672 5289536 5325492 5367573 5611076 5705762 5713014 5764987  
 L20 5825767 5875299 5903900 5911144 5915255 5930807 5948058 5968157 1234094  
5999730 6038572 6049810 6052729 6115782 6116008 6219830 6219830  
6253252 6263338 6314558 6373846 6434685 6457019 6499137 6567837  
6678697 6691304 6754898 6886159 6931423 7028287 20020099902  
20020194191 20030093577 20030097396 20030105772 20030236819  
20040001010 20040205701 20050149587 20060077993 20060126956  
4458533 4467443 4580264 4841486 5278906 6064958 6964018 6999622  
20010036318 20050182794 20050188296 20040252119 5993218 4927147  
5727788 5839902 6324631 5471471 5471472 5657314 5673318 5757913  
5844907 6389031 20030043806 4341521 4344243 4378118 4495230 4573927  
4583006 5180337 5291674 5345961 5528496 5557202 5677626 6246231  
20040200172 20060016139 5960436 5276868 5471487 5623449 5644784  
5968125 6055616 6075931 6138208 6260011 20030225998 20050223329  
20060106986 5201277 5878427 5893913 5999648 6078926 4290421 4364056

4420265 4511248 4608662 4755955 4756448 4774666 4910103 4931999  
4951241 4951238 4958159 4984183 5195147 5208905 5311488 5321801  
5361131 5392432 5483623 5485613 5505557 5574439 5696918 5752053  
5763308 5787283 5812794 5832225 5841145 5897647 5944533 6070168  
6201959 6240406 6259629 6438036 20050253335 3848710 3816709 4057342  
4243319 4243270

*DB=PGPB,USPT,USOC; PLUR=NO; OP=OR*

<input type="checkbox"/> L19	L18 and (runtime or run-time)	0
<input type="checkbox"/> L18	6987813.pn.	1
<input type="checkbox"/> L17	L16 and (n near (bit or bits))	13
<input type="checkbox"/> L16	(bit adj1 stealing)	77
<input type="checkbox"/> L15	L4 and (address same word same n same (bit or bits))	33
<input type="checkbox"/> L14	L3 and L13	4
<input type="checkbox"/> L13	(address same word same n same (bit or bits))	6799
<input type="checkbox"/> L12	L11 and (address same n same (bit or bits))	8
<input type="checkbox"/> L11	L3 and (n near (bit or bits))	27
<input type="checkbox"/> L10	L9 and (n near (bit or bits))	2
<input type="checkbox"/> L9	L3 and (header with pointer with (bit or bits))	15
<input type="checkbox"/> L8	L6 and (header with pointer with (bit or bits))	7
<input type="checkbox"/> L7	L6 and (k near (bit or bits))	25
<input type="checkbox"/> L6	L5 and (n near (bit or bits))	128
<input type="checkbox"/> L5	L4 and bit\$.ab.	822
<input type="checkbox"/> L4	header.ab.	11841
<input type="checkbox"/> L3	header.ti.	2415
<input type="checkbox"/> L2	(compact near object near header)	1
<i>DB=USPT; PLUR=NO; OP=OR</i>		
<input type="checkbox"/> L1	6748402.pn.	1

END OF SEARCH HISTORY


[Home](#) | [Login](#) | [Logout](#) | [Access Information](#) | [Alerts](#) |

Welcome United States Patent and Trademark Office

**Search Results****BROWSE****SEARCH****IEEE XPLORE GUIDE**

Results for "((virtual address and bit)&lt;in&gt;metadata)"

 e-mail

Your search matched 18 of 1392165 documents.

A maximum of 100 results are displayed, 25 to a page, sorted by **Relevance in Descending order**.**» Search Options**[View Session History](#)[New Search](#)**Modify Search**

((virtual address and bit)&lt;in&gt;metadata)

 Search Check to search only within this results setDisplay Format:  Citation  Citation & Abstract view selected items [Select All](#) [Deselect All](#)**» Key**

IEEE JNL IEEE Journal or Magazine

IEE JNL IEE Journal or Magazine

IEEE CNF IEEE Conference Proceeding

IEE CNF IEE Conference Proceeding

IEEE STD IEEE Standard

1. **Space-efficient 64-bit Java objects through selective typed virtual address**  
 Venstermans, K.; Eeckhout, L.; De Bosschere, K.;  
Code Generation and Optimization, 2006. CGO 2006. International Symposium  
 26-29 March 2006 Page(s):11 pp.  
 Digital Object Identifier 10.1109/CGO.2006.34  
[AbstractPlus](#) | Full Text: [PDF\(416 KB\)](#) [IEEE CNF](#)  
[Rights and Permissions](#)

2. **Opal: a single address space system for 64-bit architecture address space**  
 Chase, J.; Levy, H.; Baker-Harvey, M.; Lazowska, E.;  
Workstation Operating Systems, 1992. Proceedings., Third Workshop on  
 23-24 April 1992 Page(s):80 - 85  
 Digital Object Identifier 10.1109/WWOS.1992.275684  
[AbstractPlus](#) | Full Text: [PDF\(440 KB\)](#) [IEEE CNF](#)  
[Rights and Permissions](#)

3. **Some Issues for single address space systems**  
 Chase, J.; Feeley, M.; Levy, H.;  
Workstation Operating Systems, 1993. Proceedings., Fourth Workshop on  
 14-15 Oct. 1993 Page(s):150 - 154  
 Digital Object Identifier 10.1109/WWOS.1993.348157  
[AbstractPlus](#) | Full Text: [PDF\(316 KB\)](#) [IEEE CNF](#)  
[Rights and Permissions](#)

4. **Distributed persistent object system with uniform representation of point garbage collection**  
 Yamamoto, K.; Inohara, S.; Miyazawa, H.; Uehara, I.; Hara, M.; Masuda, T.;  
System Sciences, 1996. Proceedings of the Twenty-Ninth Hawaii International Conference on  
 Volume 1, 3-6 Jan. 1996 Page(s):12 - 21 vol.1  
 Digital Object Identifier 10.1109/HICSS.1996.495442  
[AbstractPlus](#) | Full Text: [PDF\(964 KB\)](#) [IEEE CNF](#)  
[Rights and Permissions](#)

5. **Distributed operating systems based on a protected global virtual address**  
 Carter, J.B.; Cox, A.L.; Johnson, D.B.; Zwaenepoel, W.;  
Workstation Operating Systems, 1992. Proceedings., Third Workshop on  
 23-24 April 1992 Page(s):75 - 79  
 Digital Object Identifier 10.1109/WWOS.1992.275685

10/19/2023

<http://ieeexplore.ieee.org/search/searchresult.jsp?query1=&scope1=metadata&op1=and&que...> 8/17/06

[AbstractPlus](#) | Full Text: [PDF\(336 KB\)](#) IEEE CNF  
[Rights and Permissions](#)

- 6. **Using virtual addresses as object references**  
Chase, J.; Levy, H.; Tiwary, A.;  
[Object Orientation in Operating Systems, 1992, Proceedings of the Second International Workshop on](#)  
24-25 Sept. 1992 Page(s):245 - 248  
Digital Object Identifier 10.1109/IWOOS.1992.252974  
[AbstractPlus](#) | Full Text: [PDF\(316 KB\)](#) IEEE CNF  
[Rights and Permissions](#)
- 7. **Page-Level Behavior of Cache Contention**  
Tambat, S.; Vajapeyam, S.;  
[Computer Architecture Letters, IEEE](#)  
Volume 1, Issue 1, Jan. 2002 Page(s):9 - 9  
Digital Object Identifier 10.1109/L-CA.2002.9  
[AbstractPlus](#) | Full Text: [PDF\(200 KB\)](#) IEEE JNL  
[Rights and Permissions](#)
- 8. **Using a distributed single address space operating system to support multicore computing**  
Skousen, A.; Miller, D.;  
[System Sciences, 1999, HICSS-32, Proceedings of the 32nd Annual Hawaii International Conference on](#)  
Volume Track8, 5-8 Jan. 1999 Page(s):10 pp.  
Digital Object Identifier 10.1109/HICSS.1999.773069  
[AbstractPlus](#) | Full Text: [PDF\(496 KB\)](#) IEEE CNF  
[Rights and Permissions](#)
- 9. **Truly online paging with locality of reference**  
Fiat, A.; Mendel, M.;  
[Foundations of Computer Science, 1997, Proceedings., 38th Annual Symposium on](#)  
20-22 Oct. 1997 Page(s):326 - 335  
Digital Object Identifier 10.1109/SFCS.1997.646121  
[AbstractPlus](#) | Full Text: [PDF\(824 KB\)](#) IEEE CNF  
[Rights and Permissions](#)
- 10. **A microprocessor design for multilevel security**  
Clifton, D.B.; Fernandez, E.B.;  
[Aerospace Computer Security Applications Conference, 1988, Fourth](#)  
12-16 Dec. 1988 Page(s):194 - 198  
Digital Object Identifier 10.1109/ACSAC.1988.113440  
[AbstractPlus](#) | Full Text: [PDF\(428 KB\)](#) IEEE CNF  
[Rights and Permissions](#)
- 11. **Virtual memory support for multiple page sizes**  
Khalidi, Y.A.; Talluri, M.; Nelson, M.N.; Williams, D.;  
[Workstation Operating Systems, 1993, Proceedings., Fourth Workshop on](#)  
14-15 Oct. 1993 Page(s):104 - 109  
Digital Object Identifier 10.1109/WWOS.1993.348164  
[AbstractPlus](#) | Full Text: [PDF\(424 KB\)](#) IEEE CNF  
[Rights and Permissions](#)
- 12. **Solemn: Solaris emulation mode for Sparc Sulima**  
Clarke, B.;  
[Simulation Symposium, 2004, Proceedings., 37th Annual](#)  
18-22 April 2004 Page(s):64 - 71  
Digital Object Identifier 10.1109/SIMSSYM.2004.1299466

[AbstractPlus](#) | Full Text: [PDF\(1374 KB\)](#) IEEE CNF  
[Rights and Permissions](#)

- 13. **Virtual page tag reduction for low-power TLBs**  
Petrov, P.; Oraloglu, A.;  
[Computer Design, 2003. Proceedings, 21st International Conference on](#)  
13-15 Oct. 2003 Page(s):371 - 374  
Digital Object Identifier 10.1109/ICCD.2003.1240921  
[AbstractPlus](#) | Full Text: [PDF\(270 KB\)](#) IEEE CNF  
[Rights and Permissions](#)
  
- 14. **Using a single address space operating system for distributed computing performance**  
Skousen, A.; Miller, D.;  
[Performance Computing and Communications Conference, 1999. IPCCC '99, International](#)  
10-12 Feb. 1999 Page(s):8 - 14  
Digital Object Identifier 10.1109/IPCCC.1999.749414  
[AbstractPlus](#) | Full Text: [PDF\(660 KB\)](#) IEEE CNF  
[Rights and Permissions](#)
  
- 15. **Don't Use the Page Number, but a Pointer to It**  
Seznec, S.;  
[Computer Architecture, 1996. 23rd Annual International Symposium on](#)  
22-24 May 1996 Page(s):104 - 104  
Digital Object Identifier 10.1109/ISCA.1996.10025  
[AbstractPlus](#) | Full Text: [PDF\(976 KB\)](#) IEEE CNF  
[Rights and Permissions](#)
  
- 16. **Supporting Reference And Dirty Bits In SPUR's Virtual Address Cache**  
Wood, D.A.; Katz, R.H.;  
[Computer Architecture, 1989. The 16th Annual International Symposium on](#)  
28 May - 1 June, 1989 Page(s):122 - 130  
[AbstractPlus](#) | Full Text: [PDF\(868 KB\)](#) IEEE CNF  
[Rights and Permissions](#)
  
- 17. **SPARC64: a 64-b 64-active-instruction out-of-order-execution MCM processor**  
Williams, T.; Patkar, N.; Shen, G.;  
[Solid-State Circuits, IEEE Journal of](#)  
Volume 30, Issue 11, Nov. 1995 Page(s):1215 - 1226  
Digital Object Identifier 10.1109/4.475709  
[AbstractPlus](#) | Full Text: [PDF\(1656 KB\)](#) IEEE JNL  
[Rights and Permissions](#)
  
- 18. **A low-power 2.5-GHz 90-nm level 1 cache and memory management unit**  
Haigh, J.R.; Wilkerson, M.W.; Miller, J.B.; Beatty, T.S.; Strazdus, S.J.; Clark, L.  
[Solid-State Circuits, IEEE Journal of](#)  
Volume 40, Issue 5, May 2005 Page(s):1190 - 1199  
Digital Object Identifier 10.1109/JSSC.2005.845971  
[AbstractPlus](#) | [References](#) | Full Text: [PDF\(1160 KB\)](#) IEEE JNL  
[Rights and Permissions](#)



USPTO

[Subscribe \(Full Service\)](#) [Register \(Limited Service, Free\)](#) [Login](#)
**Search:**  The ACM Digital Library  The Guide



THE ACM DIGITAL LIBRARY

[Feedback](#) [Report a problem](#) [Satisfaction survey](#)

## Terms used

[virtual address](#) and [bit](#) and [pointer](#) and [lock](#) and [encode](#)

Found 39,010 of 184,245

Sort results  
by
 [Save results to a Binder](#)[Try an Advanced Search](#)Display  
results
 [Search Tips](#)[Try this search in The ACM Guide](#) [Open results in a new window](#)

Results 1 - 20 of 200

Result page: [1](#) [2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [8](#) [9](#) [10](#) [next](#)

Best 200 shown

Relevance scale

- 1 ["Topologies"—distributed objects on multicomputers](#)



Karsten Schwan, Win Bo

 May 1990 **ACM Transactions on Computer Systems (TOCS)**, Volume 8 Issue 2

Publisher: ACM Press

Full text available: [pdf\(3.83 MB\)](#)Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#), [review](#)

Application programs written for large-scale multicomputers with interconnection structures known to the programmer (e.g., hypercubes or meshes) use complex communication structures for connecting the applications' parallel tasks. Such structures implement a wide variety of functions, including the exchange of data or control information relevant to the task computations and/or the communications required for task synchronization, message forwarding/filtering under program control, and so on ...

- 2 [A comparison of the object-oriented features of Ada 95 and Java](#)



Benjamin M. Brosgol

 November 1997 **Proceedings of the conference on TRI-Ada '97**

Publisher: ACM Press

Full text available: [pdf\(2.41 MB\)](#)Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)

- 3 [A dynamic network architecture](#)



Sean W. O'Malley, Larry L. Peterson

 May 1992 **ACM Transactions on Computer Systems (TOCS)**, Volume 10 Issue 2

Publisher: ACM Press

Full text available: [pdf\(401.43 KB\)](#)Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#), [review](#)

Network software is a critical component of any distributed system. Because of its complexity, network software is commonly layered into a hierarchy of protocols, or more generally, into a protocol graph. Typical protocol graphs—including those standardized in the ISO and TCP/IP network architectures—share three important properties; the protocol graph is simple, the nodes of the graph (protocols) encapsulate complex functionality, and the topology of the graph ...

**Keywords:** compositability, dynamic configuration, reuse

10/19/2023

**4 A formal description of the UNIX operating system** Thomas W. Doeppner, Alessandro Glaçalone August 1983 **Proceedings of the second annual ACM symposium on Principles of distributed computing PODC '83**

Publisher: ACM Press

Full text available:  pdf(916.14 KB)Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

In this paper we discuss our approach to a formal description of the UNIX operating system [Rit78a] [Rit78b] [Tho78], using Milner's Calculus of Communicating Systems (CCS) [Mil80]. The paper focuses on the problems one encounters and the decisions one has to make when describing a system such as UNIX. We believe that the problems that arise in defining such a system are much less well understood than those, for example, related to the formalization of programmin ...

**5 A framework for the assessment of operating systems for small computers** Hossein Saiedian, Munib SiddiqiApril 1996 **ACM SIGICE Bulletin**, Volume 21 Issue 4

Publisher: ACM Press

Full text available:  pdf(1.69 MB)Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

A number of high performance operating systems are now available for small computers on different hardware platforms. These operating systems offer many advanced features formerly reserved for their workstation and minicomputer counterparts. This article surveys the most widely used of such operating systems, namely OS/2, Windows NT, Linux and Macintosh System 7.5. It provides an account on the history, design objectives and evolution of these operating systems and discusses their key features, ...

**Keywords:** CP/M, DOS, Linux, Macintosh, Microcomputers, OS/2, Operating Systems, Small Computer Systems, Windows, Windows NT

**6 A high performance, universal, key associative access method** David B. LometMay 1983 **ACM SIGMOD Record , Proceedings of the 1983 ACM SIGMOD international conference on Management of data SIGMOD '83**, Volume 13 Issue 4

Publisher: ACM Press

Full text available:  pdf(1.81 MB)Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#)

A new file organization is proposed that combines the advantages of digital B-trees and extendible hashing methods into one organization that can be used universally. The method, like these predecessors, relies on digital searching. The key notions are: (i) that multipage nodes are addressed by the root and can have both data and index entries, the mix of entries changing over time; and (ii) that these nodes can be doubled with file growth and, when this occurs, data nodes at the next level of t ...

**7 A high-level abstraction of shared accesses** Peter J. KeleherFebruary 2000 **ACM Transactions on Computer Systems (TOCS)**, Volume 18 Issue 1

Publisher: ACM Press

Full text available:  pdf(183.57 KB)Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#), [review](#)

We describe the design and use of the tape mechanism, a new high-level abstraction of accesses to shared data for software DSMs. Tapes consolidate and generalize a number of

recent protocol optimizations, including update-based locks and recorded-replay barriers. Tapes are usually created by "recording" shared accesses. The resulting recordings can be used to anticipate future accesses by tailoring data movement to application semantics. Tapes-based mechanisms a ...

**Keywords:** DSM, programming libraries, shared memory, update protocols

#### 8 A history and evaluation of System R

Donald D. Chamberlin, Morton M. Astrahan, Michael W. Blasgen, James N. Gray, W. Frank King, Bruce G. Lindsay, Raymond Lorie, James W. Mehl, Thomas G. Price, Franco Putzolu, Patricia Griffiths Selinger, Mario Schkolnick, Donald R. Slutz, Irving L. Traiger, Bradford W. Wade, Robert A. Yost

October 1981 **Communications of the ACM**, Volume 24 Issue 10

Publisher: ACM Press

Full text available:  pdf(1.55 MB)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

System R, an experimental database system, was constructed to demonstrate that the usability advantages of the relational data model can be realized in a system with the complete function and high performance required for everyday production use. This paper describes the three principal phases of the System R project and discusses some of the lessons learned from System R about the design of relational systems and database systems in general.

**Keywords:** access path selection, authorization, compilation, database management systems, locking, recovery, relational model

#### 9 A Model-Based Approach for Executable Specifications on Reconfigurable Hardware

Tim Schattkowsky, Wolfgang Mueller, Achim Rettberg

March 2005 **Proceedings of the conference on Design, Automation and Test in Europe - Volume 2 DATE '05**

Publisher: IEEE Computer Society

Full text available:  pdf(174.47 KB)

Additional Information: [full citation](#), [abstract](#), [index terms](#)

UML 2.0 provides a rich set of diagrams for systems documentation and specification. Many efforts have been undertaken to employ different aspects of UML for multiple domains, mainly in the area of software systems. Considering the area of electronic design automation, however, we currently see only very few approaches, which investigate UML for hardware design and hardware/software co-design. In this article, we present an approach for executable UML closing the gap from system specification to ...

#### 10 A new page table for 64-bit address spaces

M. Talluri, M. D. Hill, Y. A. Khalidi

December 1995 **ACM SIGOPS Operating Systems Review , Proceedings of the fifteenth ACM symposium on Operating systems principles SOSP '95**, Volume 29 Issue 5

Publisher: ACM Press

Full text available:  pdf(1.97 MB)

Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)

#### 11 A new switch chip for IBM RS/6000 SP systems

Craig B. Stunkel, Jay Herring, Bulent Abali, Rajeev Sivaram

January 1999 **Proceedings of the 1999 ACM/IEEE conference on Supercomputing (CDROM)**

Publisher: ACM Press

Full text available:  pdf(177.66 KB) Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)

**12 A personal view of the personal work station: some firsts in the Fifties** 

◆ Douglas Ross

◆ January 1986 **Proceedings of the ACM Conference on The history of personal workstations**

Publisher: ACM Press

Full text available:  pdf(4.26 MB) Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)

**13 A portable sampling-based profiler for Java virtual machines** 

◆ John Whaley

◆ June 2000 **Proceedings of the ACM 2000 conference on Java Grande**

Publisher: ACM Press

Full text available:  pdf(1.01 MB) Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)

**14 A Robust Main-Memory Compression Scheme** 

◆ May 2005 **ACM SIGARCH Computer Architecture News , Proceedings of the 32nd Annual International Symposium on Computer Architecture ISCA '05**, Volume 33 Issue 2

Publisher: IEEE Computer Society, ACM Press

Full text available:  pdf(460.04 KB) Additional Information: [full citation](#), [abstract](#), [index terms](#)

Lossless data compression techniques can potentially free up more than 50% of the memory resources. However, previously proposed schemes suffer from high access costs. The proposed main-memory compression scheme practically eliminates performance losses of previous schemes by exploiting a simple and yet effective compression scheme, a highly-efficient structure for locating a compressed block in memory, and a hierarchical memory layout that allows compressibility of blocks to vary with a low fra ...

**15 A single intermediate language that supports multiple implementations of exceptions** 

◆ Norman Ramsey, Simon Peyton Jones

◆ May 2000 **ACM SIGPLAN Notices , Proceedings of the ACM SIGPLAN 2000 conference on Programming language design and implementation PLDI '00**, Volume 35 Issue 5

Publisher: ACM Press

Full text available:  pdf(900.75 KB) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

We present mechanisms that enable our compiler-target language, C--, to express four of the best known techniques for implementing exceptions, all within a single, uniform framework. We define the mechanisms precisely, using a formal operational semantics. We also show that exceptions need not require special treatment in the optimizer; by introducing extra dataflow edges, we make standard optimization techniques work even on programs that use exceptions. Our approach clarifies the design s ...

**16 A survey of current object-oriented databases** 

◆ Mansour Zand, Val Collins, Dale Caviness

◆ February 1995 **ACM SIGMOD Database**, Volume 26 Issue 1

Publisher: ACM Press

Full text available:  pdf(1.44 MB) Additional Information: [full citation](#), [abstract](#), [citations](#), [index terms](#)

Object-oriented concepts form a good basis for the data models required for next-generation database applications such as CAD/CAE/CASE/CAM systems, knowledge-based systems, multimedia, etc. Many object-oriented databases are available commercially or are being developed by industry or academic research facilities. This paper attempts to compare some of these products using fourteen criteria. The selected criteria are major factors required for the successful design of an object-oriented database ...

**Keywords:** OOD-BMS survey, object-oriented database, object-oriented terminology

**17 A taxonomy of computer program security flaws**

 Carl E. Landwehr, Alan R. Bull, John P. McDermott, William S. Choi  
September 1994 **ACM Computing Surveys (CSUR)**, Volume 26 Issue 3

Publisher: ACM Press

Full text available:  pdf(3.81 MB)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#), [review](#)



An organized record of actual flaws can be useful to computer system designers, programmers, analysts, administrators, and users. This survey provides a taxonomy for computer program security flaws, with an Appendix that documents 50 actual security flaws. These flaws have all been described previously in the open literature, but in widely separated places. For those new to the field of computer security, they provide a good introduction to the characteristics of security flaws and how they ...

**Keywords:** error/defect classification, security flaw, taxonomy

**18 A Tree Structured Architecture for semantic gap reduction**

 Arieh Plotkin, Daniel Tabak  
September 1983 **ACM SIGARCH Computer Architecture News**, Volume 11 Issue 4

Publisher: ACM Press

Full text available:  pdf(938.46 KB) Additional Information: [full citation](#), [abstract](#), [references](#)



The article proposes a new Tree-Structured-Architecture (TSA). The TSA is object-oriented, implements the notions of capability-based-addressing and the single-level-store, and it is particularly designed to narrow the semantic gap. It encourages modular programming and directly supports the concepts of tasks and inter-task communication, making it particularly suitable for multiprocessing and multiprogramming implementation. The TSA is implemented on a multi-resource, distributed, matrix-structu ...

**19 A unified vector/scalar floating-point architecture**

 N. P. Jouppi, J. Bertoni, D. W. Wall  
April 1989 **ACM SIGARCH Computer Architecture News , Proceedings of the third international conference on Architectural support for programming languages and operating systems ASPLOS-III**, Volume 17 Issue 2

Publisher: ACM Press

Full text available:  pdf(1.25 MB)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)



In this paper we present a unified approach to vector and scalar computation, using a single register file for both scalar operands and vector elements. The goal of this architecture is to yield improved scalar performance while broadening the range of vectorizable applications. For example, reduction operations and recurrences can be expressed in vector form in this architecture. This approach results in greater overall performance for most applications than does the approach of emphasizin ...

**20** [Accent: A communication oriented network operating system kernel](#)

Richard F. Rashid, George G. Robertson

◆ December 1981 **Proceedings of the eighth ACM symposium on Operating systems principles**

Publisher: ACM Press

Full text available:  [pdf\(1.01 MB\)](#)Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

Accent is a communication oriented operating system kernel being built at Carnegie-Mellon University to support the distributed personal computing project, Spice, and the development of a fault-tolerant distributed sensor network (DSN). Accent is built around a single, powerful abstraction of communication between processes, with all kernel functions, such as device access and virtual memory management accessible through messages and distributable throughout a network. In this paper, specif ...

**Keywords:** Distributed computation, Inter-process communication, Network, Networking, Operating systems, PERQ, Paging, UNIX, VAX, Virtual memory

Results 1 - 20 of 200

Result page: [1](#) [2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [8](#) [9](#) [10](#) [next](#)

The ACM Portal is published by the Association for Computing Machinery. Copyright © 2006 ACM, Inc.

[Terms of Usage](#) [Privacy Policy](#) [Code of Ethics](#) [Contact Us](#)Useful downloads:  [Adobe Acrobat](#)  [QuickTime](#)  [Windows Media Player](#)  [Real Player](#)